

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-320201

(43)Date of publication of application : 04.12.1998

(51)Int.Cl.

G06F 9/38

G06F 7/00

H03K 19/177

(21)Application number : 09-131834

(71)Applicant : CANON INC

(22)Date of filing : 22.05.1997

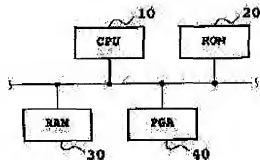
(72)Inventor : TANAKA SADAHIRO

(54) COMPUTER SYSTEM, INFORMATION PROCESSING METHOD AND RECORDING MEDIUM

(57)Abstract:

**PROBLEM TO BE SOLVED:** To unnecessitate a manual capacitance check by discriminating the arithmetic processing capacitance of programming gate array(PGA) through an information processor and selecting either the array or the processor for executing operation.

**SOLUTION:** A ROM 20 stores a program for executing function operation through a PGA 40, program for executing the same function operation through a CPU, and program for selecting any one of these programs. Besides, a RAM 30 is provided with a table showing the block of PGA 40 at a position to be used. When an instruction for replying the remaining capacitance of prescribed function is received, while referring to this table, a CPU 10 acquires the remaining capacitance of PGA 40. Next, the CPU 10 automatically discriminates whether or not the PGA 40 can provide the capacitance required for executing the prescribed function and at the time of positive discrimination, the PGA 40 executes that function operation but in case of negative discrimination, that function operation is executed on the side of CPU 10.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

特開平10-320201

(43) 公開日 平成10年(1998)12月4日

(51) Int.CI.<sup>8</sup> 識別記号  
 G 0 6 F 9/88 3 7 0  
 7/00  
 H 0 3 K 19/177

F I  
 G 0 6 F 9/88 3 7 0 C  
 H 0 3 K 19/177  
 G 0 6 F 7/00 E

審査請求 未請求 請求項の数12 O L (全 10 頁)

(21) 出願番号 特願平9-131834

(22) 出願日 平成9年(1997)5月22日

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 田中 貞治

東京都大田区下丸子3丁目30番2号 キヤ

ノン株式会社内

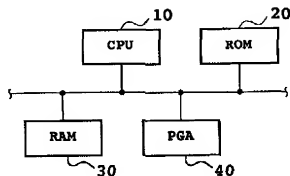
(74) 代理人 弁理士 谷 義一 (外1名)

(54) 【発明の名称】 コンピュータシステムおよび情報処理方法ならびに記録媒体

(57) 【要約】

【課題】 プログラミングターゲットレイのブロックの手作業の割付をなくす。

【解決手段】 PGAのブロック配置を示すテーブルをRAM30上に作成し、CPU10はこのテーブルを参照して、演算に必要なブロックを確保が可能かを判定する。確保できる場合には、PGA40により演算を実行し、確保できない場合には、CPU10側で演算を行う。



## 【特許請求の範囲】

【請求項1】 プログラミングゲートアレイおよび情報処理プロセッサを有し、該情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するコンピュータシステムにおいて、前記情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理と同一内容の第2の演算処理を前記情報処理プロセッサにより実行するための第2のプログラムを記憶した記憶手段と、前記第1の演算処理に要する容量が前記プログラミングゲートアレイに残存するか否かを判定する判定手段と、前記情報処理プロセッサにより実行すべきプログラムとして肯定判定が得られた場合には前記記憶手段の第1のプログラムを選択し、否定判定が得られた場合には前記記憶手段の第2のプログラムを選択するプログラム選択手段とを具えたことを特徴とするコンピュータシステム。

【請求項2】 請求項1に記載のコンピュータシステムにおいて、前記プログラミングゲートアレイに対してガベージコレクションを施す手段をさらに有し、前記判定手段の判定処理に先立って前記ガベージコレクションを施すことを特徴とするコンピュータシステム。

【請求項3】 請求項1に記載のコンピュータシステムにおいて、前記プログラミングゲートアレイに対してガベージコレクションを施す手段をさらに有し、前記判定手段の判定結果が否定判定となった場合には、前記ガベージコレクションを施した後、前記判定手段の判定を行い、その判定結果に応じて、前記第1プログラムおよび前記第2プログラムのいずれかを選択することを特徴とするコンピュータシステム。

【請求項4】 プログラミングゲートアレイおよび情報処理プロセッサを有し、該情報処理プロセッサの制御により前記プログラミングゲートアレイによる演算処理を実行するコンピュータシステムにおいて、前記情報処理プロセッサはマルチタスク処理を実行可能であり、複数のタスクに対応させて前記演算処理を行う前記プログラミングゲートアレイ内のブロックの配置を予め定めておき、

該配置を示すテーブル情報を記憶した記憶手段と、前記情報処理プロセッサの実行タスクの切替え毎に前記記憶手段に記憶されたテーブル情報に基づき、実行タスクに対応して前記ブロックと前記情報処理プロセッサとの間の前記プログラミングゲートアレイの入出力ラインを切替える制御手段とを具えたことを特徴とするコンピュータシステム。

【請求項5】 プログラミングゲートアレイおよび情報処理プロセッサを有するコンピュータシステムで、該情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するための情報処

理方法において、前記情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するための第1のプログラムおよび前記第1の演算処理と同一内容の第2の演算処理を前記情報処理プロセッサにより実行するための第2のプログラムを前記コンピュータシステム内に記憶し、前記第1の演算処理に要する容量が前記プログラミングゲートアレイに残存するか否かを前記情報処理プロセッサにより判定し、前記情報処理プロセッサにより肯定判定が得られた場合には実行すべきプログラムとして前記第1のプログラムを選択し、否定判定が得られた場合には実行すべきプログラムとして前記第2のプログラムを選択することを特徴とする情報処理方法。

【請求項6】 請求項5に記載の情報処理方法において、前記プログラミングゲートアレイに対してガベージコレクションを施した後、前記情報処理プロセッサによる容量判定を行うことを特徴とする情報処理方法。

【請求項7】 請求項5に記載の情報処理方法において、前記情報処理プロセッサの判定結果が否定判定となった場合には、前記プログラミングゲートアレイに対してガベージコレクションを施した後、前記情報処理プロセッサによる容量判定を行い、その判定結果に応じて、前記第1プログラムおよび前記第2プログラムのいずれかを選択することを特徴とする情報処理方法。

【請求項8】 プログラミングゲートアレイおよび情報処理プロセッサを有するコンピュータシステムで、該情報処理プロセッサの制御により前記プログラミングゲートアレイによる演算処理を実行するための情報処理方法において、

前記情報処理プロセッサはマルチタスク処理を実行可能であり、複数のタスクに対応させて前記演算処理を行う前記プログラミングゲートアレイ内のブロックの配置を予め定めておき、

該配置を示すテーブル情報を前記コンピュータシステム内に記憶しておき、前記情報処理プロセッサの実行タスクの切替え毎に前記テーブル情報に基づき、実行タスクに対応して前記ブロックと前記情報処理プロセッサとの間の前記プログラミングゲートアレイの入出力ラインを切替えることを特徴とする情報処理方法。

【請求項9】 記録媒体、プログラミングゲートアレイおよび情報処理プロセッサを有するコンピュータシステムで、前記記録媒体に記録されたプログラムを前記情報処理プロセッサが実行することにより前記プログラミングゲートアレイの演算処理を制御する記録媒体において、前記情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するための第1のプログラムおよび前記第1の演算処理と同一

10

20

30

40

50

内容の第2の演算処理を前記情報処理プロセッサにより実行するための第2のプログラムを前記コンピュータシステム内に記憶しておき、前記プログラムは、前記第1の演算処理に要する容量が前記プログラミングゲートアレイに残存するか否かを前記情報処理プロセッサにより判定する第3の処理手順と、

前記情報処理プロセッサにより肯定判定が得られた場合には実行すべきプログラムとして前記第1のプログラムを選択し、否定判定が得られた場合には実行すべきプログラムとして前記第2のプログラムを選択する第2の処理手順とを具えたことを特徴とする記録媒体。

【請求項10】 請求項9に記載の記録媒体において、前記第1の処理手順に先立って、前記プログラミングゲートアレイに対してガベージコレクションを施す処理手順をさらに具えたことを特徴とする記録媒体。

【請求項11】 請求項9に記載の記録媒体において、前記第1の処理手順の判定結果が否定判定となった場合には、前記プログラミングゲートアレイに対してガベージコレクションを施す第3の処理手順と、該第3の処理手順の後、前記第1の演算処理に要する容量が前記プログラミングゲートアレイに残存するか否かを前記情報処理プロセッサにより判定する第4の処理手順と該第4の処理手順の判定結果に応じて、前記第1プログラムおよび前記第2プログラムのいずれかを選択する第5の処理手順をさらに具えたことを特徴とする記録媒体。

【請求項12】 記録媒体、プログラミングゲートアレイおよび情報処理プロセッサを有するコンピュータシステムで、前記記録媒体に記録されたプログラムを前記情報処理プロセッサが実行することにより前記プログラミングゲートアレイの演算処理を制御する記録媒体において、前記情報処理プロセッサはマルチタスク処理を実行可能であり、複数のタスクに対応させて前記演算処理を行う前記プログラミングゲートアレイ内のブロックの配置を予め定めおき、前記プログラムは、該配置を示すテーブル情報を前記コンピュータシステム内に記憶しておき、

前記情報処理プロセッサの実行タスクの切替えを検出する処理手順と、該検出に応じて前記テーブル情報に基づき、実行タスクに対応して前記ブロックと前記情報処理プロセッサとの間の前記プログラミングゲートアレイの入出力ラインを切替える処理手順とを具えたことを特徴とする記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、プログラミングゲートアレイ（PGA）を使用して情報処理を行うコンピュータシステムおよび情報処理方法ならびに記録媒体に関する。

【0002】

10

【従来の技術】PGAはAND（アンド）、OR等の論理演算を行うゲート回路を複数個任意に接続することが可能な情報処理回路であり、このためプログラマブルゲートアレイとも呼ばれている。PGAはハードウェアICデザインへのテストとか、少量生産品用のゲートアレイの代替品とかに使用されているが、一部の関数計算をPGAに実行させ、他の関数計算をCPUやデジタルプロセッサ（DSP）のソフト処理により実行するコンピュータシステムが提案されている。

【0003】

10

【発明が解決しようとする課題】従来、この種のコンピュータシステムは、PGAは、使用可能なゲート数に制限があるために、予め定めた個数の関数の演算を実行させるようにプログラムすることが通常であり、使用する関数の計算のために使用するゲートの個数等を予め計数し、PGAの使用可能な個数（容量）に合致するように関数の内容を定めなければならない。また、このような作業は手作業で行われるため、PGAを使用するコンピュータシステムでは、実際に稼動するまでの準備に多大

20

な時間を有するという解決すべき問題があった。【0004】そこで、本発明の目的は、上述の点に鑑みて、事前に手作業によるPGAの容量チェックを行う必要をなくし、情報処理の実行に係る制約を緩和することができるコンピュータシステムおよび情報処理方法ならびに記録媒体を提供することにある。

【0005】

30

【課題を解決するための手段】このような目的を解決するために請求項1の発明は、プログラミングゲートアレイおよび情報処理プロセッサを有し、該情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するコンピュータシステムにおいて、前記情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するための第1のプログラムおよび前記第1演算処理と同一内容の第2の演算処理を前記情報処理プロセッサにより実行するための第2のプログラムを記憶した記憶手段と、前記第1の演算処理に要する容量が前記プログラミングゲートアレイに残存するか否かを判定する判定手段と、前記情報処理プロセッサにより実行すべきプログラムとして肯定判定が得られた場合には前記記憶手段の第1のプログラムを選択し、否定判定が得られた場合には前記記憶手段の第2のプログラムを選択するプログラム選択手段とを具えたことを特徴とする。

40

【0006】請求項2の発明は、請求項1に記載のコンピュータシステムにおいて、前記プログラミングゲートアレイに対してガベージコレクションを施す手段をさらに有し、前記判定手段の判定処理に先立って前記ガベージコレクションを施すことを特徴とする。

50

【0007】請求項3の発明は、請求項1に記載のコンピュータシステムにおいて、前記プログラミングゲート

アレイに対してガベージコレクションを施す手段をさらに有し、前記判定手段の判定結果が否定判定となった場合には、前記ガベージコレクションを施した後、前記判定手段を行い、その判定結果に応じて、前記第1プログラムおよび前記第2プログラムのいずれかを選択することを特徴とする。

【0008】請求項4の発明は、プログラミングゲートアレイおよび情報処理プロセッサを有し、該情報処理プロセッサの制御により前記プログラミングゲートアレイによる演算処理を実行するコンピュータシステムにおいて、前記情報処理プロセッサはマルチタスク処理を実行可能であり、複数のタスクに対応させて前記演算処理を行う前記プログラミングゲートアレイ内のブロックの配置を予め定めおき、該配置を示すテーブル情報を記憶した記憶手段と、前記情報処理プロセッサの実行タスクの切替え毎に前記記憶手段に記憶されたテーブル情報に基づき、実行タスクに対応して前記ブロックと前記情報処理プロセッサとの間の前記プログラミングゲートアレイの入出力ラインを切替える制御手段とを具えたことを特徴とする。

【0009】請求項5の発明は、プログラミングゲートアレイおよび情報処理プロセッサを有するコンピュータシステムで、該情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するための情報処理方法において、前記情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するための第1のプログラムおよび前記演算処理と同一内容の第2の演算処理を前記情報処理プロセッサにより実行するための第2のプログラムを前記コンピュータシステム内に記憶し、前記第1の演算処理に要する容量が前記プログラミングゲートアレイに残存するか否かを前記情報処理プロセッサにより判定し、前記情報処理プロセッサにより肯定判定が得られた場合には実行すべきプログラムとして前記第1のプログラムを選択し、否定判定が得られた場合には実行すべきプログラムとして前記第2のプログラムを選択することを特徴とする。

【0010】請求項6の発明は、請求項5に記載の情報処理方法において、前記プログラミングゲートアレイに対してガベージコレクションを施した後、前記情報処理プロセッサによる容量判定を行うことを特徴とする。

【0011】請求項7の発明は、請求項5に記載の情報処理方法において、前記情報処理プロセッサの判定結果が否定判定となった場合には、前記プログラミングゲートアレイに対してガベージコレクションを施した後、前記情報処理プロセッサによる容量判定を行い、その判定結果に応じて、前記第1プログラムおよび前記第2プログラムのいずれかを選択することを特徴とする。

【0012】請求項8の発明は、プログラミングゲートアレイおよび情報処理プロセッサを有するコンピュータ

システムで、該情報処理プロセッサの制御により前記プログラミングゲートアレイによる演算処理を実行するための情報処理方法において、前記情報処理プロセッサはマルチタスク処理を実行可能であり、複数のタスクに対応させて前記演算処理を行う前記プログラミングゲートアレイ内のブロックの配置を予め定めおき、該配置を示すテーブル情報を前記コンピュータシステム内に記憶しておき、前記情報処理プロセッサの実行タスクの切替え毎に前記テーブル情報に基づき、実行タスクに対応して前記ブロックと前記情報処理プロセッサとの間の前記プログラミングゲートアレイの入出力ラインを切替えることを特徴とする。

【0013】請求項9の発明は、記録媒体、プログラミングゲートアレイおよび情報処理プロセッサを有するコンピュータシステムで、前記記録媒体に記録されたプログラムを前記情報処理プロセッサが実行することにより前記プログラミングゲートアレイの演算処理を制御する記録媒体において、前記情報処理プロセッサの制御により前記プログラミングゲートアレイによる第1の演算処理を実行するための第1のプログラムおよび前記第1の演算処理と同一内容の第2の演算処理を前記情報処理プロセッサにより実行するための第2のプログラムを前記コンピュータシステム内に記憶しておき、前記プログラムは、前記第1の演算処理に要する容量が前記プログラミングゲートアレイに残存するか否かを前記情報処理プロセッサにより判定する第1の処理手順と、前記情報処理プロセッサにより肯定判定が得られた場合には実行すべきプログラムとして前記第1のプログラムを選択し、否定判定が得られた場合には実行すべきプログラムとして前記第2のプログラムを選択する第2の処理手順とを具えたことを特徴とする。

【0014】請求項10の発明は、請求項9に記載の記録媒体において、前記第1の処理手順に先立って、前記プログラミングゲートアレイに対してガベージコレクションを施す処理手順をさらに具えたことを特徴とする。

【0015】請求項11の発明は、請求項9に記載の記録媒体において、前記第1の処理手順の判定結果が否定判定となった場合には、前記プログラミングゲートアレイに対してガベージコレクションを施す第3の処理手順と、該第3の処理手順の後、前記第1の演算処理に要する容量が前記プログラミングゲートアレイに残存するか否かを前記情報処理プロセッサにより判定する第4の処理手順と該第4の処理手順の判定結果に応じて、前記第1プログラムおよび前記第2プログラムのいずれかを選択する第5の処理手順をさらに具えたことを特徴とする。

【0016】請求項12の発明は、記録媒体、プログラミングゲートアレイおよび情報処理プロセッサを有するコンピュータシステムで、前記記録媒体に記録されたプログラムを前記情報処理プロセッサが実行することによ

り前記プログラミングゲートアレイの演算処理を制御する記録媒体において、前記情報処理プロセッサはマルチタスク処理を実行可能であり、複数のタスクに対応させて前記演算処理を行う前記プログラミングゲートアレイ内のブロックの配置を予め定めておき、前記プログラムは、該配置を示すテーブル情報を前記コンピュータシステム内に記憶しておき、前記情報処理プロセッサの実行タスクの切替えを演出する処理手順と、該演出に応じて前記テーブル情報に基づき、実行タスクに対応して前記ブロックと前記情報処理プロセッサとの間の前記プログラミングゲートアレイの入出力ラインを切替える処理手順とを具えたことを特徴とする。

【0017】

【発明の実施の形態】以下、図面を参照して本発明の実施の形態を詳細に説明する。

【0018】（第1実施形態）図1はコンピュータシステムの一実施の形態のシステム構成を示す。図1において、CPU10はROM20に格納されたシステムプログラム（オペレーティングシステムとも呼ばれる）に従って、システム制御を実行するほか、各種の関数計算を実行するための後述のアプリケーションプログラムを格納する。

【0019】なお、システムプログラムの中には四則演算や論理演算あるいは特定の関数演算を行うための従来から周知のプログラムが含まれているものとする。本実施の形態の特徴は、PGA40がある関数を実行するために必要な容量（未使用のゲート数または、ブロック）を提供できるかをCPU10により自動判定し、肯定判定（容量あり）の場合は、その関数の演算をPGA40に実行させ、否定判定（容量なし）の場合にCPU10側でその関数演算を実行するようにしたことの特徴がある。

【0020】ROM20は、CPU10が実行するシス

```
int (search)();
... 中略
if (Flag==PGA_MALLOCC(2))
{
    then {
        setup_hardware(flag);
        search = search_hardware;
    }
    else
        search = search_software;
}
search();
if (Flag==true then
    free_hardware(flag)
int setup_hardware(int flag);
{
    chip_upload(flag);
    chip_IO_setup();
```

(1)

(2)

(3)

＊システム制御のためのシステムプログラムおよび各種の関数演算を規定したアプリケーションプログラムを格納する。このアプリケーションプログラムの中には、PGA40をプログラム（関数を実行するブロック配置およびゲートの接続を決定する）として、PGA40により関数演算を実行するためのプログラム、同じ関数演算をCPU10により実行するためのプログラム、これらプログラムを選択するためのプログラム（図2、図4）が含まれている。上記システムプログラムやアプリケーションプログラムをハードディスク記憶装置のような大容量記憶装置に記憶することも可能である。

【0021】RAM30は、CPU10に対する入出力情報を一時記憶する。PGA40は従来から既知のものを使用でき、CPU10の指示するプログラム（ゲート構成）で論理演算を行う。PGA40の使用箇所のブロックを示すテーブル（FPGA\_Mallocテーブル、図3参照）がRAM30内に設けられ、CPU10は、たとえば、FPGA\_Malloc関数のような残量を応答せよの命令を受けると、このテーブルを参照することによりPGA40の残存容量を取得する。なお、PGAのプログラミングに応じて上記FPGA\_Mallocテーブルの内容が更新されることは言うまでもない。

【0022】以下、本発明に係るPGA制御について図2および図3を参照して説明する。なお、図2は、演算処理をPGA40で行うか、CPU10で行うかを決定するための処理手順を示す。図3はFPGA\_MallocテーブルとPGA40内で確保されるブロックとの対応関係を示す。

【0023】図2の処理手順に対応するプログラムスク립トの一例を以下に示す。

【0024】

```

9
int *x;
int ret_val;
ret_val=no;
for (x=FPGA_Table; x<FPGA_Table_End;x++){
    if (*x==yes) then{
        if (x->Lengthof(int)*x_length >FPGA_Table_End)
            then return;
        for (y=x;y<x_length;y++){
            if (*x !=yes) then {
                x=y;
                break;
            }
        }
        ret_val=x;
        return
    }
}

```

関数の演算を実行する際に、CPU10は図2の処理手順を読み出して実行する。図2において、CPU10はPGA40の残量の問い合わせを行う。より具体的には実行コマンドがFPGA\_MALLOCC関数であることを識別すると、CPU10はこのコマンドで指示されたブロック数、上述のスク립ト例では2×2のブロックがPGA40において、確保可能かをFPGA\_Mallocテーブルを参照してその可否を決定する(ステップS10→S20)。上述のスク립トではFlagがその可否を現し、Flagがtrue(真)の場合に、確保可能を表す。この一連の処理を定義したプログラム命令に上記スク립トの中で符号(1)を付している。

【0025】図2の判定において要求ブロックが確保可能な場合には、PGA40による演算(PGA40を使用するためのプログラムを起動)を指示し(ステップS20→S30)、確保不可の場合には、CPU10によるソフトウェア演算(CPU10による演算を規定したソフトウェアプログラムの起動)を指示する(ステップS20→S40)。

【0026】PGA40を使用するためのプログラムを起動するための、上記スク립ト中のプログラム命令に符号(2)を付し、CPU10による演算を規定したプログラムの起動を指示するプログラム命令には符号(3)を付している。

【0027】上述のMalloc関数はさらにブロックの確保まで行う命令であり、図3に示すようにPGA40内に2×2ブロックが確保され、FPGA\_Mallocテーブルも新規に確保されたブロックが使用中として記載される。

【0028】ちなみに上述のスク립トの例では符号(3)のプログラム命令以降には以下の処理が規定されている。すなわち、ハードウェア(PGA40)を使用する場合には、ハードウェアを開放し、割り当てられた

位置にサッチ用ハードウェアをロードし、チップ1/Oをこのハードウェアに接続し、テーブル参照によりPGA40の空きエリアを探し、空きブロックの値を設定することが上記スク립トで定義されている。

【0029】したがって、ユーザは、予め、PGA40を使用するソフトウェアプログラムと、同一の演算内容で、CPU10により実行するプログラムをコンピュータシステム内の記録媒体に用意しておき、上述の処理手順にしたがって、プログラムを選択すればよい。この処理手順によれば、ユーザがPGA40用のプログラムを複数用意する場合でも、ユーザは、各プログラムについてブロックがPGA40の最大使用可能ブロック数を越えないように配慮すればよく、各プログラム毎にブロック数の割り当てを行う必要がない。

【0030】(第2実施形態) 次にPGA40内で複数の演算を実行させるために有効な第2実施例を説明する。システム構成を図1の第1実施形態と同様とすることができる。第2実施形態におけるシステム制御手順を図4に示す。PGA40内の割り当てられた使用ブロックの配置が図5の(A)で示すような配置となっていた場合、2×2ブロックをPGA40上でとることはできない。そこで、第2実施形態では、PGA40が要求ブロックの確保が可能かの判定で否定判定が得られた場合(図4のステップS100→S110のNo判定)にはガベージコレクションを施して(ステップS120)、図5の符号(B)に示すように使用ブロックを詰めて、空きブロックの空間を大きくする。

【0031】この後、PGA40が要求ブロックの確保が可能かの判定を行うと、上述の2×2の要求ブロックを確保できることとなる(ステップS140→S130、図5(C)参照)。ガベージコレクションを施しても要求ブロックが得られない場合には、CPU10によるソフトウェア実行処理を選択することとなる(ステ

ップS140→S150)。

\*を以下に示す。

【0032】このような処理を行うためのスクリプト例\*

【0033】

```
if (Flag=FPGA_MALLOCC(2) then
  ...PGA処理...
else {
  FPGA_Collect().....ガベージコレクションの指示
  if (Flag=FPGA_MALLOCC(2) then
    {
      ...PGA処理...
    }
  else {
    return(error_Flag1);
  }
}
```

(第3実施形態) CPU10がマルチタスク処理を実行する場合のPGA40側の関数切り換え処理を次に説明する。マルチタスク(複数のプログラム等を平行しながらCPUに実行させる処理)によりタスクに対応させて異なる関数を演算実行する場合、図7の符号(A)で示すようにPGA40の1番、3番のラインはタスク1で使用し、1番、2番のラインをタスク2で使用するようなことがある。

【0034】そこで、CPU10の実行する処理においてタスクの切り換えが起きたときにはPGA40のブロックに対するCPU10との間の入出力ラインの配線をCPU10の指示で切替える。このためのCPU10の処理手順を図6に示す。なお、タスク番号に対応させて使用する結線番号を記載した管理テーブル(LINE\_Ma11ocテーブル)がRAM30内に予め記憶されているものとする。

【0035】CPU10は既存のOS(オペレーティングシステム)によりタスクの切替えを検知すると(ステップS200)、RAM30内のLINE\_Ma11ocテーブルを参照し、タスク番号に対応する結線番号を取得して、PGA40の結線を接続し直す(ステップS210→S220→S230)。

【0036】これによりマルチタスクを使用する複雑な情報処理にも対応して、PGA40を効率的に使用し、各種の関数処理をPGA40側で実行することができ

る。

【0037】

【発明の効果】以上説明したように、請求項1、5、9の発明によれば、演算を行う場合にプログラミングゲートアレイに残存容量があれば、プログラミングゲートアレイにより演算が行われ、その演算を高速に行うことができる。また、プログラミングゲートアレイに残存容量がなくても、演算が情報処理プロセッサにより行われるので、演算処理支障をきたすことがない。このため、ユ

ーザは、プログラミングゲートアレイのブロックの割り当てを気にせず、自由な関数を定義できる。

【0038】請求項2、6、10の発明によれば、プログラミングゲートアレイの容量の残存判定に先立って、ガベージコレクションを実行することで、プログラミングゲートアレイの使用可能な空き領域を拡大して、空き領域の使用効率を高めることができる。

【0039】請求項3、7、11の発明によれば、残存容量判定で否定判定が得られたときにガベージコレクションを施し、再び残存容量判定を行う。

【0040】これにより、残存容量がある場合には第1回目の判定で肯定判定が得られるので、ただちにプログラミングゲートアレイにより演算を開始できるとともに、第1回目の判定で否定判定が得られた場合にもガベージコレクションを施すことで、空き領域が増え、プログラミングゲートアレイにより演算を実行させるためのブロックを確保することが可能になる。

【0041】請求項4、8、12の発明によれば、情報処理プロセッサがマルチタスクによりプログラミングゲートアレイを使用することができ、これにより、多種多様な情報処理を提供できる。

【図面の簡単な説明】

【図1】本発明実施の形態のシステム構成を示すブロック図である。

【図2】CPU10の処理手順を示すフローチャートである。

【図3】PGAのブロック配置を示す説明図である。

【図4】CPU10の処理手順を示すフローチャートである。

【図5】PGAのブロック配置を示す説明図である。

【図6】CPU10の処理手順を示すフローチャートである。

【図7】PGAのブロック配置を示す説明図である。

【符号の説明】



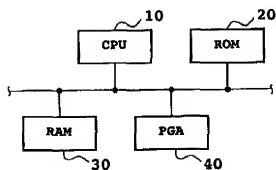
10 CPU

\* 30 RAM

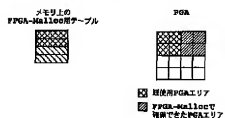
20 ROM

\* 40 プログラミングゲートアレイ (PGA)

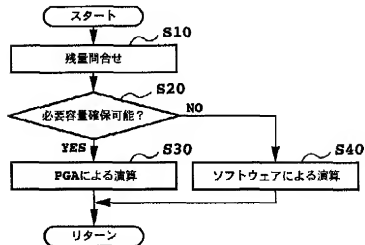
【図1】



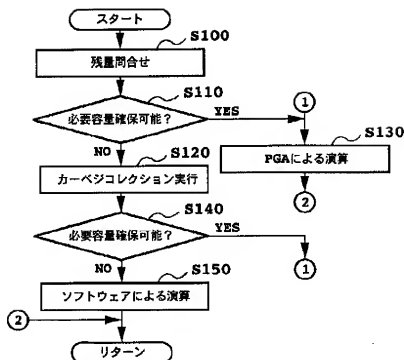
【図3】



【図2】



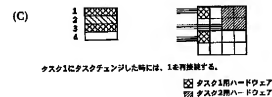
【図4】



【図5】



【図7】



【図6】

